CLUBE

Z80

Agosto/85

N.º 35

Coordenadores: ANA PAULA • ALEXANDRE SOUSA • J. MAGALHAES • AV.ª BOAVISTA, 832-2.º T. • TELEF. 65127 • 4100 PORTO

NESTE NÚMERO

Edição: Clube Z80

Fotocomposição: Fotomecânica Mabreu/Porto

Impressão: Ramos dos Santos & Ca., Lda./Porto

Tiragem: 500 exemplares, Agosto 1985

INTRODUÇÃO AO CÓDIGO MÁQUINA

Autor: FERNANDO PRECES
SACAVÉM

(Cont. dos números anteriores)

A instrução OUT (Porta) A, indica ao microprocessador que deve colocar uma cópia do byte DATA, contendo do registo A, na porta especificada pelo segundo byte da instrução. Nas restantes instruções deste grupo a porta é especificada pelo conteúdo do registro C.

A instrução IND é semelhante em comportamento à instrução LDD. Um DATA recolhido na porta especificada pelo conteúdo do registro C é colocado em memória no endereço apontado por HL. Os registros B e HL são depois decrementados.
A instrução INDR é uma instrução de transferência de bloco semelhante à LDDR. Ela transfere um bloco DATA da porta endereçada por C para os endereços de memória apontados sucessivamente por HL. Os registros B e HL são decrementados até o conteúdo de B ser zero.

A instrução INI é uma instrução equivalente à LDI. Ela transfere um DATA da porta endereçada por C para o endereço apontado por HL. O registro B é decrementado e HL incrementado.

A instrução INIR é uma instrução de transferência de bloco semelhante à LDIR. Ela transfere um bloco de DATA da porta endereçada por C, para os endereços apontados por HL. O registro B é decrementado até zero e o registro HL incrementado.

A instrução OUTD é a inversa da IND. Ela envia o conteúdo do endereço apontado por HL, para a porta endereçada pelo registro C. Os registros B e HL são decrementados.
A instrução OTDR é a inversadaA INDR. Ela envia o conteúdo endereçado por HL para a porta indicada pelo registro C. O registro HL é decrementado até o registro B ser zero.
A instrução OUTI é a inversa da INI. Ela coloca o conteúdo do endereço apontado por HL na porta endereçada pelo registro C. O registro B é decrementado e HL incrementado.
A instrução OTIR é a inversa da INIR. Ela envia o conteúdo do endereço apontado por HL para a porta endereçada pelo registro C. O registro B é decrementado e HL incrementado.
Exemplos extraídos da ROM do ZX81
Instrução IN — A rotina de pesquisa do teclado dá-nos um exemplo de como é utilizada uma instrução IN a (c).

Como já fizemos referência neste texto, no sistema do ZX81, o teclado está ligado a 8 linhas do BUS de endereços e a pressão de cada uma das teclas emite sinais diferentes nas 5 linhas do BUS de DATA.

Assim para estabelecer as coordenadas exactas de uma tecla, o teclado é pesquisado por um ciclo que por 8 vezes e caminhos diferentes, verifica a tecla primida.
Esta pesquisa que se efectua a velocidade vertiginosa é processada 50 vezes por segundo.

Por oito vezes o endereço no BUS de endereços é mudado usando uma instrução RLC B e um bit «marca» orienta a evolução do ciclo. Inicialmente o bit 0 é o único que se encontra a zero e ao entrar no Cany flag provoca uma situação de loops (ciclo ou cadeia fechada).
As instruções envolvidas:

Endereços	Mnemónicas	Comentários
699	LD HL, 65535	Parâmetros de iniciação
702	LD BC, 65278	
705	IN A, (c)	1.ª entrada
707	OR 1	
709	OR 224	
711	LD D, A	Desenvolvimento do valor individual para cada tecla
712	CPL	
713	CP 1	
715	SBC A, A	
716	OR B	
717	AND L	
718	LD HL, A	
719	LD A, H	
720	AND D	
721	LD H, A	
722	RLC B	Rotação que prepara o loop
724	IN A, (c)	
726	JR C 709	Salta até à entrada da «marca» no Cany

No retorno o registro HL retém as coordenadas da tecla primida.
Instrução OUT — No ZX81, a fim de se tornar visível no écran a entrada de cada byte, quando um programa está a ser carregado duma cassete, um **eco** do byte copiado na porta 255 lhe é adicionado para aumentar a sua largura.

Endereços	Mnemónicas	Comentários
940	IN A (255)	Copia o byte
942	OUT (255) A	adiciona o eco

Ensaio 1 (ZX81) — a rotina que vamos utilizar produz som que pode ser gravado em cassete. Variando o conteúdo do registro B e introduzindo pequenas alterações na rotina, poder-se-á gravar música na cassete. Experimente.

	Mnemónicas	Comentários
	LD B 255	Frequência do impulso
	LD A 255	Largura do impulso
LOOP	OUT (255) A	Saída para a cassete
	DJNZ LOOP	Controlo do contador
	RET	

```
100 FOR N = 1 TO 300
110 LET K = USR X
120 NEXT N
```

Em que (X) é o endereço de arranque da rotina em cm.
Ensaio 2 — Spectrum

Quando se começa a programar em código máquina, cometem-se erros que normalmente são devidos à inexperiência ou ao entusiasmo. Enquanto que no Basic o programa monitor nos apresenta o erro, não aceitando a respectiva linha até este ser corrigido, em CM não temos tal protecção e na maior parte dos casos perdemos o programa.

Assim, quando o leitor tiver dúvidas sobre o funcionamento correcto de algumas instruções que prentenda inserir no seu programa, poderá incluir um conjunto de 4 instruções que provocam o retorno ao Basic, caso prima a tecla BREAK. Isto não protege o programa dum **crash,**mas protege-o dum **loop** contínuo.

Mnemónicas	Comentários
LD A 127 IN A (254)	Verifica se BREAK é primido
RRA	bit 0 para o Cany
RET NC	

Ensaio 3 — Screptum

No Basic existem 2 comandos (IN e OUT) um quanto semelhantes às instruções Assembler IN a (c) e OUT a (c). Fatalmente serão mais lentas e não requerem especificações muito detalhadas.

Ambas devem ser seguidas de um endereço que o Z80A reconhece como pertencente a um pequeno grupo de endereços de entrada que lêem o teclado e a saída «EAR». Note que os endereços que vou representar em baixo, não devem ser confundidos com endereços de memória.
Tal como no ZX81, oito endereços são necessários para ligar o teclado ao Z80A, sendo cada um responsável pela detecção de meia fila de teclas (5 teclas). Deve relembrar o capítulo 23 do Manual do Spectrum.

Endereços das portas	Teclas	correspondentes
65278 (1.ª 1/2 fila)	CAPS SHIFT	a V
65022 (2.ª	A	a G
64510 (3.ª	Q	a T
63486 (4.ª	I	a 5
61438 (5.ª	O	a 6
57342 (6.ª	P	a y
49150 (7.ª	ENTER	a H
32766 (7.ª	SPACE	a B

Cada tecla tem um código de referência dentro da 1/2 fila aonde pertence.

Como o bit 6 pertence à leitura «EAR» por razões técnicas para as quais ainda não arranjei uma resposta, pode nalguns Spectruns entrar com o valor 1 e noutros entrar com o valor 0.

Assim, o valor de tecla primida apanhado na porta 254, pode ser por exemplo para a tecla A e código 254 nalguns casos e noutros o código 190.
Códigos:

Decimal	Binário	Decimal	Binário	Ordem de Tecla
254 —	11111110	190 —	10111110	1.ª
253 —	11111101	189 —	10111101	2.ª
251 —	11111011	187 —	10111011	3.ª
247 —	11110111	183 —	10110111	4.ª
239 —	11101111	175 —	10101111	5.ª

Quando não há tecla primida.
255 — 11111111 190 — 10111111

Para que o leitor possa saber como reage o seu Spectrum, deve executar este pequeno teste.

```
10 REM (teclas A e G)
20 LET X = IN 65022
30 IF X = 255 OR X = 191 THEN GOTO 20
40 IF X = 254 OR X = 190 THEN PRINT X;
   «Tecla A»; GOTO 20
50 IF X = 239 OR X = 175 THEN PRINT X;
   «Tecla G»; GOTO 20
```

Prima as teclas A ou G e verifique os códigos obtidos. Idêntico ensàio pode ser formulado em Assembler para testarmos os valores obtidos ao primir uma tecla.

Mnemónicas		Comentários
START:	LD BC 32766	Endereço 1/2 fila (SPACE a B)
	IN A (c)	Copia código na porta
	CP 255	Tecla não primida? versão 1
	JR Z, START	
	CP 191	Tecla não primida? versão 2
	JR Z, START	
	LD B, 0	
	LD C, A	Código da tecla
	RET	

PRINT USR X (endereço do CM)
No retorno ao Basic obtém o código da tecla primida, (1/2 fila ENTER a B).
Ensaio 4 — Spectrum
Numa instrução OUT para a porta 254 o byte DATA pode conter 3 informações distintas.
Os bits 0, 1 e 2 dão informação de cor (8 cores) para o BORDER, o bit 3 informação para a ficha do «MIC», o bit 4 informação para o anti-falante e os bits 5, 6 e 7 não têm aplicação directa.
Vamos examinar algumas dessas aplicações. A primeira rotina vai produzir uns leves ruídos no anti-falante, semelhante do produzido ao primir uma tecla. Digamos que reproduzirá o eco da tecla.

Mnemónicas		Comentários
	LD B 255	
	LD A 0	Bit 4 a zero
LOOP 1:	OUT (254) A	Cone anti-falante não mexe
	DJNZ LOOP 1	
	LD B 255	
	LD A 16	Bit 4 a 1
LOOP 2:	OUT (254) A	Desloca cone do anti-falante
	DJNZ LOOP 2	
	RET	

A segunda rotina vai afectar a cor do BORDER no écran e em simultâneo o anti-falante.

Mnemónicas		Comentários
	LD B 255	
LOOP 1:	PUSH BC	Protege o valor de B
	LD A B	
LOOP 2:	OUT (254) A	
	DJNZ LOOP 2	
	POP BC	
	DJNZ LOOP 1	Tira 1 a B e salta
	RET	

A terceira rotina simula uma explosão.
Variando os parâmetros contidos em D, e C, poderemos obter efeitos bastante interessantes.

Mnemónicas		Comentários
	LD HL 0	
	LD C 0	
LOOP 1:	LD D 32	
LOOP 2:	LD A (HL)	
	AND 24	Para evitar a variação de tons no écran
	OUT (254) A	
	LO BC	
	DJNZ LOOP	
	INC HL	
	DEC D	
	JR NZ LOOP 2	
	N C	
	JR NZ LOOP 1	
	RET	

O DATA integrado na instrução AND, é o decimal 24 que em Binário é igual a 00011000.

Como pode verificar os bits 0, 1 e 2, que nos dão a cor do BORDER no écran, são mantidos a zero, pelo que vamos obter quando rolamos na rotina um BORDER de cor negra Se pretende obter um BORDER multicor, pode substituir a instrução AND 24 por OR 24 que vai continuar a permitir a excitação do anti-falante pelo bit 4 e ao mesmo tempo deixar passar toda a sequência de números possível (0 a 7 — 8 cores do Spectrum), pelos bits 0, 1 e 2.

(Continua no próximo número)

UMA VIAGEM PELO INTERIOR DO SPECTRUM

(Continuação do número anterior)

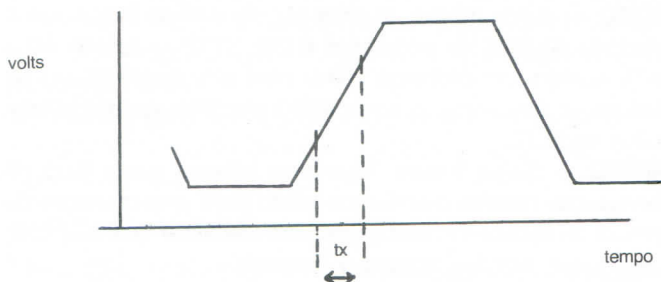
2 — **O cérebro do Spectrum**

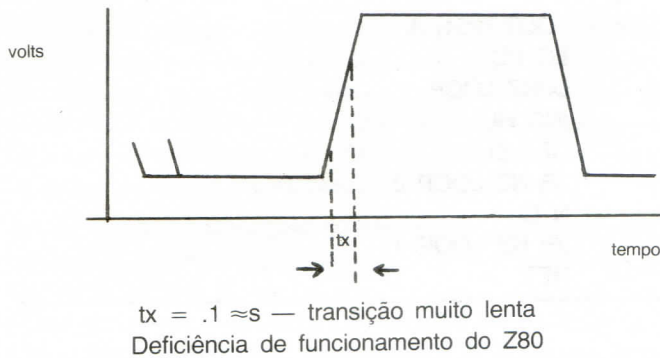
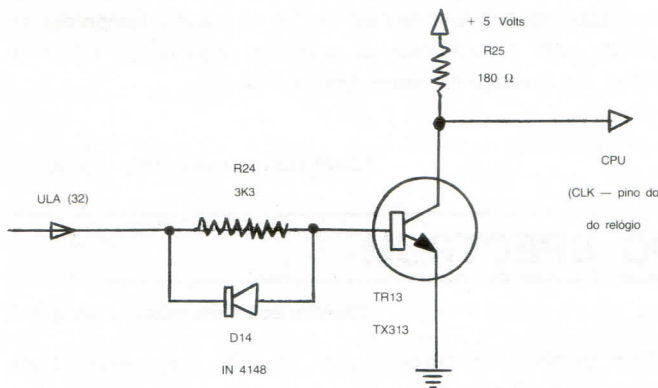
Um comprido integrado de caixa plástica, alberga o processador central, um Z80A, que está ligado aos circuitos do computador pelos seus 40 pinos. Ele, vai assegurar na íntegra as seguintes funções:

a) a leitura de dados na memória;
b) a introdução de dados na memória;
c) a leitura de dados nas portas Input/Output (i/O;
d) a introdução de dados nas portas (I/O);
e) a execução de operações aritméticas e lógicas entre dados.

O trabalho que o Z80A executa é determinado pelas instruções que sequencialmente vai retirando da memória. Há **150 tipos diferentes** de instruções máquina que o Z80A compreende e pode executar.
Muitos computadores utilizam o integrado Z80 como processador central. A família Z80 é composta por três tipos de integrados (o Z80, o Z80A e muito recentemente, o Z80B). Todos eles falam a mesma linguagem utilizando palavras de 8 bits, e a diferença mais flagrante situa-se na velocidade operacional.

Z80 — Gama dos 2,5 megaciclos
Z80A — Gama dos 4 megaciclos
Z80B — Gama dos 6 megaciclos

Existem no mercado outros microprocessadores que funcionam com palavras de 16 ou 32 bits. Estes, sem dúvida, são mais rápidos comparados com os Z80, visto executarem maior quantidade de trabalho em tempo igual, mas dado o seu preço não são utilizados em computadores caseiros.
O sinal do relógio que sincroniza o funcionamento do Z80A, oscila à frequência de 3,5 megaciclos. Então, todas as operações efectuadas por este processador (daqui para a frente vamos chama-lo CPU — Unidade de Processamento Central) estão referenciadas com o sinal-relógio, excepto quando actuar na informação de VÍDEO (sinal para o TV).
Contam-se por centenas os circuitos individuais do CPU que têm de **arrancar** em simultâneo, no desempenho de determinada tarefa. Daí a necessidade de um rigoroso sincronismo, que apenas é conseguido, se as transições entre o mínimo e o máximo de sinal proveniente do relógio, sejam suficientemente rápidas para assegurar essa simultaneadade.

tx = .03 μs — transição aceitável
Razoável funcionamento do Z80

tx = .1 ≈s — transição muito lenta
Deficiência de funcionamento do Z80

O Sinal-relógio a 3,5 megaciclos enviado pela ULA (Unidade de Lógica Aritmética) tem um instante de transição demasiado lento. Para corrigir essa inclinação o circuito de TR3, assegurar que a transição de sinal no pino de relógio do CPU seja efectuado no tempo ideal (comutação rápida).
Componentes electrónicos que formam o circuito de comutação rápida.

O BUS de Endereços, pinos A0 a A15 do CPU, são saídas tristate (3 estados). Estas saídas podem tomar o valor 0, 1, ou ficar flutuantes. (Ver texto INTRODUÇÃO À LINGUAGEM MÁQUINA — Revista n.º 18 de Março de 1984).
Quando flutuantes, essas saídas podem ser utilizadas por outros dispositivos sem **incomodar** o CPU.
Um BUS de 16 linhas pode trabalhar com 65536 endereços ou com 256 portas que são utilizadas pelo CPU para comunicar com o exterior.
O BUS de DATA, pinos D0 a D7 do CPU, são 8 linhas bidireccionais também tristate, utilizadas para Entrada/Saída de dados.
Estas linhas interligam o CPU, às memórias ou aos periféricos.

IORQ — Saída tristate. O sinal por ela emitido indica que a metade do BUS de endereços (LOW BYTE — Linhas A0 a A7), contém um endereço válido para uma operação I/O de leitura ou de escrita. A saída IORQ identifica **apenas** operações IN/OUT.
MREQ — Saída tristate. Este sinal informa que o BUS de endereços contém o endereço válido para uma operação de leitura ou escrita. O Sinal existe para identificar qualquer operação que envolva acesso à memória.
Os sinais **IORQ** e **MREQ não podem** ser activadas ao mesmo tempo.
MI — Ciclo Máquina. Este sinal informa que o CPU está correctamente preparado para executar uma instrução enviada pela memória. **MI** também aparece em conjunto com IORQ para identificar uma autorização de interrupção de ciclo.

RDREAD — Saída tristate. Informação que o CPU aguarda uma leitura de memória ou uma operação I/O. Este sinal é usado com o endereço seleccionado para colocar o DATA no BUS de DADOS do CPU.

WD Write — Saída tristate. Indica que o BUS de DADOS do CPU tem um DATA preparado para ser colocado na localização endereçada, ou ainda um DATA I/O para uma porta seleccionada.
A memória utiliza este sinal para armazenar o DATA retirado do BUS de DADOS.

RFSR Refresh — Este sinal indica que os 7 bue o BUS de DADOS do CPU tem um DATA preparado para ser colocado na localização endereçada, ou ainda um DATA I/O para uma porta seledos de serem recarregadas com as tensões bit correspondentes aos valores armazenados em cada célula).
O CPU tem de activar as células de memória para que estas **não esqueçam** os DADOS armazenados.
Quando este sinal é activado, as restantes 8 linhas do BUS de Endereços indicarão o conteúdo do Registro I do CPU, que completam o endereço.

HALT — Esta saída indica que o CPU está a executar uma instrução HALT. Esta acção que será apreciada mais adiante é uma espécie de corte de relações com todos os componentes que o cercam.

WAIT — Sinal usado pelas memórias ou por qualquer dispositivo I/O para informar CPU que o sistema ainda não está preparado para uma transferência de DATA. O CPU aguarda que a situação se normalize.

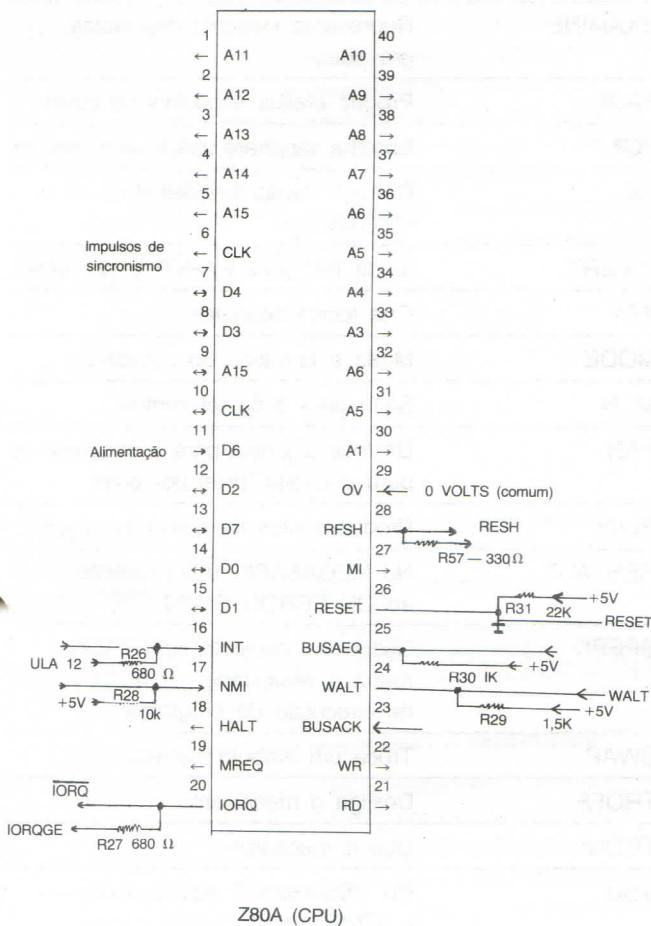
INT — Pedido de interrupção — Este sinal é gerado por dispositivos externos para indicar ao CPU uma sequência sitivo I/O para informar CPU que o sistema ainda não está preparado para uma transferência de DATA. O CPU aguarda que a situação se nocões estiver **desarmado** o CPU aceitará a interrupção pedida.

NMI — Ordem de interrupação — Este sinal é sempre aceite pelo CPU que salta para o endereço 102 da ROM, assim que termina a execução da instrução em curso.

RESET — Este sinal força o CPU a saltar para o endereço 0 da ROM, apagando toda a RAM ao recomeçar a rotina de iniciação. Isto é conseguido, colocando a armadura activa de C27 ao potencial comum.

BUSREQ — Sinal externo que solicita ao CPU o **aluguer** temporário do BUS de Endereços, do BUS de DADOS e dos sinais de Control. Esta operação deixará o CPU indiferente à transferência de DADOS entre dispositivos externos.

BUSACK — Este sinal de saída é a aceitação do CPU ao pedido efectuado pelo sinal BUSREQ e que indica ao dispositivo externo que passa daí em diante a ter controlo sobre todos os BUSES.
A maioria dos sinais apresentados serão mais adiante detalhados quando forem apresentados os respectivos circuitos.

Z80A (CPU)

(Continua no próximo número)

MEGABASIC

YS MEGABASIC é um programa em código máquina que aumenta as capacidades do seu Spectrum 48K.
Tem 22K de memória e ao mesmo tempo possui caracteristicas de um computador cinco vezes mais caro que o Spectrum — janelas, caracteres de diferentes tamanhos e formas, procedimentos, assim como uma série de novos comandos.
Além de tudo isto, o programa foi feito para acabar com o maior problema do Spectrum — ter os comandos escritos.
Este programa é a única maneira de engrandecer o Spectrum, a acima de tudo não envolve o gasto de avultosas somas de dinheiro. Se deseja fazer uma viagem através da versatilidade do seu Spectrum, veja este maravilhoso programa.

Teclado mágico

Veja como o seu teclado trabalha com o programa MEGABASIC.
As teclas trabalham como num computador "normal", ou seja, dão apenas simples caracteres ASCII. Agora todos os comandos e funções terão de ser primidos letra por letra. Isto permitirá ao utilizador do YS MEGABASIC, usar facilmente os novos comandos e, é claro, um novo utilizador do Spectrum não precisa de se adaptar ao sistema tradicional do teclado.
No programa YS MEGABASIC, a tecla de SPACE é usada como uma nova tecla SHIFT, e é como devem ser utilizadas todas as novas funções do teclado.
Veja e observe, agora o seu Spectrum está provido com teclas definidas.
As teclas do topo do teclado (telas 1 a 0) podem ser programadas para criar uma string com mais de 255 caracteres.
O método utilizado envolve o uso de um segundo cursor, que pode ser movido à volta do écran, independentemente do uso habitual do cursor do INPUT.
O texto pode ser copiado do segundo cursor para o cursor do INPUT com apenas um simples toque numa tecla.
O tipo de écran do Spectrum divide-se em duas partes. O YS MEGABASIC emprega três! Existe uma secção para os comandos do INPUT e para mostrar as mensagens de erro, outro para o OUTPUT do programa e finalmente uma área para listagens automáticas.
O tamanho e a posição destas áreas do écran são completamente definidas, e a mesma área pode ser usada por 2 (ou todas as 3) funções.

Écrans de sonho

YS MEGABASIC fez coisas espectaculares pelo écran.
Agora, oferece, não menos que, três diferentes tamanhos de caracteres:

1 — 64 colunas por 24 linhas — Aqui a largura de cada caracter continua bem clara. Este tamanho de caracter é ideal para aplicações como processamento de texto.
2 — 32 colunas por 24 linhas — Este é o tamanho normal usado pelo Spectrum comum.
3 — 32 colunas por 12 linhas — Agora já pode produzir caracteres que podem ter o dobro do tamanho normal.

Outro bónus . . . quando está a escolher entre as opções 2 e 3 é possível seleccionar entre 2 posições diferentes dos caracteres; com efeito isto dá-lhe 5 formas de posicionar os caracteres no écran do seu Spectrum.
Já alguma vez pensou ter um sistema de janelas QL no seu Spectrum?
Agora existe uma forma de manter o posicionamento limitado a uma área particular do écran.
O tamanho e a localização das janelas pode ser, totalmente, definida pelo utilizador, e caracteres de diferentes tamanhos podem ser misturados dentro da mesma janela. O utilizador pode mesmo mudar de janela quando quiser; de facto o número total de janelas activas é apenas limitado pela memória disponível ou não.
Cada janela pode ser apagada ou invertida individualmente, e a janela do écran pode ser deslocada para cima, para baixo, esquerda ou direita.

Sprites espantosas

Agora, pela primeira vez, o Spectrum incorpora a sua própria rotina de Sprites.
Tudo o que tem a fazer é definir a direcção que deseja para mover a Sprite. Pode definir o que acontece quando a Sprit bate noutro objecto gráfico . . . pode parar ou continuar a mover-se.

Um Sprite é definido pelos bytes na memória — como um gráfico definido — e cada byte representa uma linha horizontal de 8 pixeis. E mais cada quadrado de 8 por 8 pixeis pode ter o seu atributo — portanto as Sprites podem ser multi-coloridas.
Com o MEGABASIC, o seu Megaspectrum dar-lhe-à 10 Sprites — embora existam formas de obter mais. Lembre-se, no entanto, que a velocidade a que o MEGABASIC corre depende do tamanho e do número de Sprites que está a usar. Quanto mais Sprites existirem e quanto maior for o seu tamanho, mais lentamente correrá o programa.
Definir Sprites é um processo muito complicado. O programa permite-lhe, em primeiro lugar, definir as Sprites no écran e codificá-las na memória como bytes. Pode então gravar estes bytes como código, que podem ser reproduzidos nos seus próprios programas.
Um bando de novos comandos voará para os seus dedos, graças ao MEGABASIC. Por exemplo existem alguns para manejar as janelas e os diferentes tamanhos dos caracteres — como MODE, CURRENT, WINDOW e CLW.
Os comandos entram no seu Megaspectrum como entraria qualquer comando normal do Basic, cada um deles sendo "escrito" letra por letra.
Todos os comandos podem entrar em programas e tanto estes como os já conhecidos podem ser misturados na mesma linha. A única diferença entre os dois é que se um novo comando requere parâmetros, então um caracter sublinhado tem de ser posto entre uma palavra de comando e o começo do argumento.
Por exemplo, o comando para imprimir o caractere «A» no écran, usando a palavra chave «VDU», seria «VDU _ 65».
Embora o sistema de teclado existente não seja standardizado e seja, normalmente, mais um embaraço que uma ajuda, tem também algumas vantagens. Por exemplo, quando o comando PRINT entra é mais rápido primir a tecla P que as 5 letras da palavra. Com o seu MEGABASIC é possível abreviar alguns comandos e funções. Notará que os comandos são executados com um ponto final, mas quando estes aparecem no écran, aparecem «escritos» por extenso. Qualquer palavra que não esteja incluída na tabela não pode ser abreviada e tem de ser escrita por inteiro.

COMANDOS DO MEGABASIC

COMANDO	COMENTÁRIO
AUTO	Faz com que o computador produza automaticamente números de linha
BACKUP	Copia files do gravador
CHANGE	Manipula os atributos da file
CLW	Limpa a janela que está no écran no momento
CURRENT	Muda a janela usada para o OUTPUT do écran
DELETE	Apaga um bloco de linhas do programa de Basic do YS MEGABASIC
DOWN	Imprime uma String no fundo do écran
EDIT	Chama uma linha
EXAMINE	Representa Headers das linhas gravadas
FADE	Produz efeitos especiais no écran
FONT	Escolhe caractere usado para imprimir
FX	Carrega várias funções do MEGABASIC
INVERT	Muda INK para PAPER e vice-versa
KEY	Cria teclas definidas
MODE	Muda o tamanho do caractere
MON	Salta para o painel central
PAN	Desloca a janela para a esquerda ou para a direita, pixel por pixel
PLAY	Produz efeitos de som complexos
RESTART	No MEGABASIC é equivalente ao ON ERROR GOTO
SPEED	Usado em conjunto com TRON reduz a velocidade de execução do programa
SWAP	Troca um atributo por outro
TROFF	Desliga o mecanismo
TRON	Liga o mecanismo
VDU	No MEGABASIC equivalente a PRINT CHR$
WINDOW	Define o tamanho e a localização da janela no écran

Abreviações dos comandos do MEGABASIC

A.TTR	GO S.UB	PE.EK
BE.EP	G.O TO	PL.OT
B.IN	I.NKEY$	P.OINT
BO.RDER	INP.UT	PR.INT
BR.IGHT	INV.ERSE	RA.NDOMIZE
CH.R$	L.EN	RE.AD
CI.RCLE	LI.NE	RES.TORE
CLE.AR	LL.IST	RET.URN
CL.OSE	LP.RINT	R.ND
C.ODE	LO.AD	SA.VE
CON.TINUE	ME.RGE	S.CREEN$
DA.TA	M.OVE	ST.R$
D.EF FN	NE.XT	T.AB
DR.AW	N.OT	TH.EN
ER.ASE	OP.EN	U.SR
E.XP	OV.ER	V.AL$
FL.ASH	PA.PER	VE.RIFY
F.ORMAT	PAU.SE	

PODER DE PERSONALIZAÇÃO

Outro pequeno bónus, é a possibilidade de personalizar o MEGABASIC para responder às suas necessidades — basta juntar subrotinas com nome.

O sistema é similar aos procedimentos de outros Basics, excepto de que não pode usar variáveis locais.
Os parâmetros podem ser passados para as subrotinas como normalmente e a subrotina é terminada por uma instrução de RETURN do Basic. Pode activá-la fazendo entrar o seu nome (da rotina), junto com outros parâmetros requeridos. A palavra «PROC» não é requerida. A subrotina com um nome torna-se, realmente, como qualquer comando do Basic. Por exemplo:

```
10 OCHARA _ z
20 FOR A = Z TO 255
30 VDU _ A
40 NEXT A
50 RETURN
```

Este exemplo mostra como a subrotina é definida. Pode activá-la fazendo entrar «CHARA _ 165» que resulta nos caracteres de 165 até 255, representados no écran.

As subrotinas podem ser chamadas de qualquer lado no programa MEGABASIC... até se podem chamar a si próprias.

RESULTADOS DO EXAME

Um pequeno programa de um painel frontal está escondido nas profundidades do MEGABASIC... pequeno, contido em si e, é claro, escrito em código máquina.

Isto permite-nos examinar o trabalho interno do Mega Spectrum, possibilitando ao utilizador examinar, alterar memória e registros, executar programas em código máquina e mover e preencher blocos da memória. O programa trabalha apenas com números hexadecimais e pode activá-lo sempre que quiser, bastando para isso a tecla de SPACE e a tecla F ao mesmo tempo.

Pode ser activado de dentro de um programa, com a execução do comando MON.

PRÁTICA DE PROCEDIMENTOS

Como estruturar programas com o MEGABASIC?

O MEGABASIC ajuda os seus utilizadores a estruturarem os seus próprios programas. Um programa do MEGABASIC pode ser escrito em secções estruturadas, chamadas «Procedimentos». A principal vantagem destes procedimentos é que cada um deles pode ser testado antes de ser incorporado no programa principal.
Outro aspecto que torna estes procedimentos muito úteis é que pode nomear cada um deles. Usando este sistema para estruturar os seus programas, cada procedimento terá um nome relacionado com o mesmo, que deverá descrever a sua verdadeira proposta dentro do programa principal.
Assim quando uma secção, em particular, tiver de ser executada, tudo o que tem a fazer para chamar o procedimento é usar o nome da secção numa linha do programa — oposto ao uso do GO SUB que não lhe dá nenhumas indicações do que se passa.

ORGANIZAÇÃO

No MEGABASIC o começo de um procedimento é definido por um símbolo O, seguido do seu título especifico — depois, se for requerido, será seguido por um caracter sublinhado e uma lista de parâmetros.
Lembre-se que nenhum comando do Basic pode ser usado para especificar os procedimentos, portanto não chame a um procedimento «PRINT»... use a imaginação.
Uma série de Basics que incluem o uso de procedimentos, também permitem o uso de variáveis «locais». Com este sistema, variáveis usadas dentro de um procedimento, não afectam o valor da variável com o mesmo nome fora desse procedimento. Isto não é possível com o MEGABASIC, portanto tenha cuidado quando escolher os nomes para as variáveis dos procedimentos. O fim de um procedimento é definido pelo comando ENDPROC; quando um programa encontra o comando ENDPROC, o processo é continuado a partir da exposição depois da chamada do procedimento. O nome do procedimento pode ser colocado no fim do comando ENDPROC para dar maior legibilidade ao programa.
EX: «ENDPROC _ DISPLAY» — define o fim do procedimento «DISPLAY».
Os parâmetros podem também passar a procedimentos.

EMPILHAR OS EXTRAS

Outra característica do MEGABASIC, é a inclusão da estrutura REPEAT... UNTIL. O comando REPEAT marca o princípio de um loop, o UNTIL é o fim do mesmo. O comando UNTIL é sempre seguido por uma expressão — se a expressão for falsa, é feito um salto até à expressão após o último comando REPEAT e, se for verdadeira, a execução do programa continuará a partir da expressão a seguir ao comando do UNTIL.

```
9000 ODISPLAY_A$
9010 DIM Z$(64)
9020 LET Z = (64-LEN A$)/2
9030 LET Z$(Z TO ) = A$
9040 PRINT 'CHR$ 1;Z$
9050 ENDPROC _ DISPLAY
```

Este exemplo pode ser usado para imprimir uma string no écran, no centro duma linha, usando o procedimento «DISPLAY _ A$». Por exemplo:
«DISPLAY _ YS MEGABASIC» é equivalente ao ZX Basic: «LET A$ = "YS MEGABASIC": GO SUB 9010». Qual dos dois prefere usar?

```
10 REPEAT
20 OUT 254, RND *7
30 UNTIL _ INKEY$ = "z"
```

Esta listagem do MEGABASIC faz o FLASH do BORDER até a tecla Z ser primida.

```
10 OUT 254,RND*7
20 IF INKEY$ π□ "z" THEN GO TO 10
```

Agora veja isto feito pelo ZX basic (equivalente) — assustador, não acha?

```
  10 MTASK-9000
  20 FONT 0
  30 PRINT "Task 1";
  40 GO TO 30
9000 CURRENT _ 0
9010 FONT 1
9020 PRINT "Task 2";
9030 GO TO 9020
```

Este programa permite imprimir 'Task 1' na janela 2, usando o caracter comum do ZX Spectrum. Permite imprimir "Task 2" na janela 0 usando o caracter alternativo.

Os dois procedimentos e os comandos REPEAT...UNTIL usam uma pilha para guardar números de linha e de argumento. Portanto quando o procedimento é chamado, o número de linha e de argumento do comando depois da chamada é posto na pilha — então, quando o comando ENDPROC é executado, o valor é removido da pilha. Quando o comando REPEAT é processado, o número de linha e de argumento são postos na pilha (algo semelhante à chamada de procedimento).
Quando UNTIL é executado, se a expressão for verdadeira, então o valor é removido da pilha e a execução do programa continua normalmente. Mas se a expressão é falsa, o valor mais elevado da pilha é copiado e é dado um salto até ao argumento correcto.
Os loops REPEAT...UNTIL e os procedimentos podem ficar juntos. Existe espaço suficiente na pilha para 10 valores, mas se tentar colocar mais de 10 valores, terá a seguinte mensagem de erro "PROC stack overflow".

Estes são 3 comandos para alterar a pilha sem provocar um salto no programa:
POP, PUSH e PCLEAR. POP é usado para remover um valor da pilha; PUSH especifica o número de linha e de argumento para serem colocados na pilha; PCLEAR limpa a pilha. O comando PCLEAR deve ser sempre usado no princípio de qualquer programa, involvendo o uso de procedimentos ou os loops REPEAT...UNTIL, para instruir a pilha.

AO SABOR DA CORRENTE

Existem dois comandos que podem afectar o correr do programa — MTASK e BRANCH.
MTASK dá ao Spectrum uma forma simples de 'multi-tasking'; divide o programa em duas partes; executa a linha a partir da primeira parte e a seguir executa a linha a partir da segunda. Este 'multi-tasking' continua até que uma parte do programa termine ou se desligue.
MTASK é sempre seguido por um número que define a primeira linha, da segunda parte do programa (a primeira parte do programa é assumida para continuar depois do comando MTASK).
Se quiser desligar tudo, faça MTASK 0. As duas secções do programa têm de usar as mesmas variáveis, mas cada uma delas tem acesso a janelas e caracteres diferentes.
O segundo comando, BRANCH faz com que uma subrotina dê um salto depois de uma linha ter terminado. Não se esqueça que MTASK tem prioridade sobre BRANCH, portanto se deseja 'multi-task', não tente fazer 'BRANCH' ao mesmo tempo.
O número depois do comando BRANCH especifica o número de linha, no qual a subrotina começa. A subrotina é terminada por um ENDPROC.
O comando BRANCH é muito útil para quando quiser apagar erros de um programa.

JANELAS E MANIPULAÇÃO DE CARACT. NO MEGABASIC

Uma "Janela" é uma área no écran na qual pode 'colocar' quaisquer caracteres. Esta área pode ter qualquer tamanho ou posição no écran.
Os utilizadores podem ter diversas janelas activas ao mesmo tempo; o limite normal é dez, sem involver o uso especial de memória. Se as janelas estão posicionadas no écran como se estivessem deitadas, descobrirá que imprimindo numa janela, destruirá o que estiver na outra. Por isso o melhor é 'ordenar' e 'estudar' o écran antes de cometer erros.
Também descobrirá que os comandos que se moverem para fora das janelas estão perdidos — as técnicas das janelas não têm o sistema que encontrará no Apple Mackintosh. Mas também existe uma grande diferença entre o ~reço de um Spectrum e de um Mackintosh.
O Spectrum normal é visto como tendo uma simples janela com um tamanho fixo, e associada a essa janela temos as seguintes características: — A cor PAPER, a cor INK, o estado BRIGHT, o estado FLASH, o estado INVERSE e o estado OVER.
Com o MEGABASIC, o Spectrum pode fazer uso de 10 janelas definidas, cada uma delas tendo as seguintes características atribuídas: — O tamanho do caracter, a cor PAPER, a cor INK, o estado BRIGHT e o estado FLASH. O caracter corrente ou FONT, o estado INVERSE e o estado OVER, pode dizer-se serem características de todas as 10 janelas e portanto referidas como características globais. Note-se que o estado FONT só tem efeito quando se selecciona caracteres normais ou com o dobro do tamanho. Caracteres com metade da largura usam um caracter especial pré-definido.

CARACTERES

MEGABASIC, pode dizer-se que, tem 64 colunas e 24 linhas. Esta característica permite que sejam usados caracteres com metade da largura no écran; caracteres com metade da largura podem ser imprimidos em qualquer coluna do écran, considerando que caracteres normais ou com o dobro do tamanho podem ser imprimidos apenas em colunas que tenham um número. É preciso notar-se que embora a resolução do caracter tenha melhorado, a resolução da cor não melhorou.
O caracter estabelecido para FONTs 1 e 2 está guardado na RAM, permitindo aos utilizadores redefinirem-no segundo a sua vontade e para os requerimentos que tiverem. Os caracteres 128 até 143 estão também guardados na RAM (o caracter normal do Spectrum). Assim, agora é possível usar o

método de construir gráficos definidos do Spectrum para obter mais de 229 UDGs — Pense só nos gráficos que poderá incluir nos seus MEGA-Programas.

CLW_(n), n — Este comando permite ao utilizador limpar uma janela de diversas maneiras. A expressão numérica estará normalmente na linha "0" para "3". Qualquer outro número será tomado pelo Megaspectrum como sendo um "3". As acções deste comando dependem muito do valor das expressões abaixo anotadas:

0 — A janela é limpa como o comando CLS no Spectrum normal.
1 — A janela é limpa com INK em vez de PAPER.
2 — A janela é invertida, e a INK torna-se em PAPER e vice-versa.
3 — Apenas os atributos são limpos. Esta última opção permite-lhe mudar a cor da janela sem limpar o conteúdo da mesma.

CURRENT _ n — Permite-lhe seleccionar qual a janela a usar para o Output. A variável n pode ter qualquer valor entre 0 e 9. Qualquer valor acima do estabelecido resultará na seguinte mensagem "Illegal Window"

DOWN _ n,n$ — Este comando imprime uma string no fundo do écran. As duas expressões numéricas dão as coordenadas segundo as quais a string seria imprimida.

FX _ n,n — Este comando permite ao utilizador controlar o funcionamento do sistema operativo do MEGABASIC. FX chama valores de "0", "1", "2" e "3", dando opção aos utilizadores de escolher as janelas para tarefas específicas do sistema. Por exemplo:

FX _ 0,n — Escolhe janela para o comando do Input
FX _ 1,n — Escolhe janela para listagens automáticas.
FX _ 2,n — Selecciona a janela usando-a como Output.
FX _ 3,n — Escolhe a janela usada pelo painel frontal.

Usando cada um destes FX, é possível usar a mesma janela para as 4 funções.

PAN_n,n — Este comando é usado para deslocar a janela para a esquerda ou para a direita, pixel por pixel. A primeira expressão define-se a janela está cheia com INK ou PAPER, e a segunda controla qual a direcção e quantos pixeis a janela pode deslocar.

SCROLL _ n,n — Este comando é similar a PAN, excepto que as deslocações do écran são controladas de cima para baixo, e não da esquerda para a direita.

WINDOW _ n,n,n,n — Este comando define o tamanho e a posição da janela. Quando o comando é executado, a posição do cursor é RESET para o canto esquerdo no topo da janela. As duas primeiras expressões numéricas definem a posição da janela — primeiro a coordenada Y, depois a coordenada X. As outras duas expressões definem a altura e o comprimento da janela respectivamente. Se alguma destas duas expressões for zero, terá uma mensagem de erro «Window too small». Se a coordenada Y mais a altura dão uma janela maior que 24 linhas, ou a coordenada X e o comprimento dão uma janela maior que 64 colunas, aparecerá a mensagem de erro "A window too large".

FONT _ n — Este comando escolhe o caracter a ser usado para o output do écran.
FONT _ 0 — Dá o caracter normal do ZX Basic
FONT _ 1 e FONT _ 2 — Seleccionam entre os dois tipos de caracter da RAM. Se o valor "n" for superior a 2, a instrução FONT _ 2 será a assumida.

MODE _ (n),n — O comando MODE possibilita o tamanho do caracter a ser usado na impressão. Se forem especificadas 2 expressões, a primeira denota a janela na qual o tamanho do caracter é definido, a segunda é o tamanho do caracter a ser seleccionado. Se usar apenas uma expressão não será feita nenhuma alteracão na janela. O valor de "n" deve ser "1", "2" ou "3"; qualquer outro valor dado significa que nenhuns caracteres serão imprimidos — uma forma mais fácil de apagar uma janela, se o desejar.

PRINT CHR$ — O tamanho dos caracteres pode também ser definido com a instrução PRINT usando códigos de controlo. Os tamanhos "1", "2" e "3" são usados para seleccionar o tamanho do caracter para a janela.
PRINT CHR$ "MEGABASIC" imprimirá a mensagem "MEGABASIC" usando caracteres com o dobro da altura.

MEGABUGFIX

«Your Spectrum» — N.° 14, Maio

As versões anteriores a 4.0, contém alguns erros de origem. Para proceder à sua correcção faça LOAD da sua versão do MEGABASIC, e a seguir introduza esta listagem. Grave-a e depois faça RUN. Se o Sepctrum fizer NEW, e porque tudo esta correcto, se der algum erro, verifique de novo a listagem.
Para gravar a nova versão do MEGABASIC faça:
SAVE "MB" CODE 44996, 20373, ou se pretender gravar em microdrive.
SAVE "m";1;"MB" CODE 4496,20373.
Quando o quiser voltar a reproduzir de novo, faça: CLEAR 44999: LOAD "CODE, ou para microdrive faça: CLEAR 44999 LOAD* "M";1;"MB" CODE. Depois de lido o MEGABASIC, executa-se automaticamente por ele mesmo.

```
1000 LET C=0
1010 FOR A=47110 TO 47367
1020 READ B: POKE A,B
1030 LET C=C+B
1040 NEXT A
1050 IF C<>27311 THEN PRINT "che
cksum error": STOP
1055 POKE 53361,195:DOKE_53362,4
7352
```

```
1060 RANDOMIZE USR 47110
1070 DATA 62,229,50,80,222,62,25
5,50,115,206,1,195,175,205,183,3
0,33,36,219,34,196,175,33,50,184
,94,35,86,35,123
1080 DATA 178,202,183,17,1,3,0,2
37,176,24,240,195,6,184,37,219,1
95,67,184,125,219,195,90,184,78,
210,195,227,184,0,0,58,20,255,16
7,202
1090 DATA 20,5,33,84,184,229,237
,115,61,92,207,49,33,0,220,195,4
0,219,33,200,174,34,54,92,205,11
0,13,175,211,254,62,6,50,72,92,5
0,143
1100 DATA 92,50,141,92,205,175,1
3,62,2,205,1,22,17,173,184,175,2
05,10,12,58,8,92,254,46,40,3,195
,173,207,6,69,33,112,23,126,35,5
0,143
1110 DATA 92,229,197,175,17,164,
184,205,10,12,193,225,16,238,195
,173,207,128,4,32,76,65,82,65,32
,132,128,22,0,0,4,89,83,32,77,69
,71,65
1120 DATA 66,65,83,73,67,13,13,3
,86,101,114,115,105,111,110,32,4
9,46,49,13,13,2,127,32,77,105,10
7,101,32,76,101,97,109,97,110,32
,49,57
1130 DATA 56,53,13,141,254,255,4
0,7,190,194,85,210,195,81,210,58
1140 DATA 52,233,167,202,179,181
,193,175,201
1150 DATA 237,91,218,251,19,19,1
9,19,19,19,19,19,19,195,117,208
```

ALGUMAS DICAS ACERCA DO MEGABASIC

"YOUR SPECTRUM-N.16"

O YS MEGABASIC dá-lhe a oportunidade de desligar o poder normalmente associado com o código máquina. O programa usa 19K de RAM, deixando 22K disponíveis para escrever no seu programa MEGABASIC. Poderá parecer que perdeu muita memória, mas verá que não irá precisar de muita para criar écrans com este programa. E a propósito, quando foi a última vez que fez um programa com 22K?
YS MEGABASIC incorpora uma simples palavra chave, quase todos os comandos devem ser abreviados. Por exemplo, pode primir RA em vez de RANDOMISE.

Algumas características

— YS MEGABASIC reconhece cerca de 35 comandos novos, permitindo-lhe fazer muito mais com os comandos do ZX BASIC.

— 64 colunas para texto, podendo ser útil para o processador de texto ou base de dados.
— Caracter com três tipos de tamanhos diferentes: (standard, dobro e metade).
— Três fontes de caracteres.
— Janelas — Pode trabalhar ao mesmo tempo com 6 janelas.
— Sprites — Fica apto a desenhar e a manipular 10 Sprites no écran ao mesmo tempo.
— Realça o som, pegando nos Beep do Spectrum e dando-lhe o tratamento MEGA.
— Procedimento — estrutura os seus programas e define novos comandos.
— Divide os caracteres em pequenos pontos.
— Com o programa MEGABASIC vai um utilitário chamado SPRITE DESIGNER, este utilitário com 11K vai ajudá-lo a criar SPRITES no Spectrum. Pode ter dez Sprites no écran ao mesmo tempo, e se for cuidadoso com a memória poderá ter mais.

MEGADEMO

Adapt. do "Your Spectrum"
por Isabel Cristina

Se ainda não é utilizador do YS MEGABASIC ou se já é mas tem ainda certas dúvidas acerca da sua utilidade, apresentamos-lhe um programa de demonstração de algumas das capacidades deste utilitário.
O programa MEGABASIC é introduzido, simplesmente através de LOAD ""; deverá antes de mais consultar as instruções que acompanham este programa. Notará que depois de introduzir o programa, se accionar a tecla do comando PRINT, lhe aparecerá só um P, o seu Spectrum está pronto a trabalhar com um teclado igual ao da maioria dos computadores. Para escrever PRINT, deverá escrever letra por letra. Poderá depois de instalado o YS MEGABASIC, introduzir o programa apresentado e gravá-lo com SAVE "nome que quiser", podendo assim guardá-lo para fazer as demonstrações acerca do tipo de caracteres, divisões feitas no écran, gráficos, som, etc.
Cada uma das linhas que compõe o programa, e a seguir descrita, para que possa perceber melhor estas instruções.

Linha 16 — Limpa todos os procedimentos. Deve ser sempre introduzida no princípio de todos os programas.
17 — Marca o princípio do programa, e faz com que ele se repita.
20 — Chama a introdução.
30 — Chama o procedimento das janelas.
40 — Chama o procedimento gráfico,
50 — Produz uma mensagem de End, quando o programa acaba.
60 — Faz com que a demonstração se repita eternamente.
1000 — Define o princípio da Introdução.
1003 — Selecciona caracter de comprimento alongado e limpa o écran.
1005 — Define a janela corrente que cobre o écran por inteiro.
1010 — Define as cores de arranque para o procedimento da introdução.
1020/1120 — Este ciclo escreve "YS MEGABASIC DEMO" no écran, em 7 cores diferen-

tes. A linha 1070 mostra como pode escrever caracteres com tamanhos diferentes, com o mesmo PRINT. A linha 1100 coloca o cursor no canto superior esquerdo do écran, mas não limpa o écran.

A linha 1075 produz um efeito continuo de som.

1130 — Escreve a mensagem rotina do écran, usando caracteres de Controlo com a expressão Print.

1140 — Desliga o efeito de som e vai produzir outro, usando o comando PLAY.

1145 — As duas últimas linhas do écran ficam azuis.

1150/1160 — Abre duas janelas na cor azul no fim do écran.

1170/1200 — Preenche o écran de azul até ao meio separando-o em duas direcções diferentes.

1204 — Manda a cor das duas linhas de Input, para o fim do écran.

1205/1250 — O écran fica a fazer Flash em 8 cores diferentes.

1260 — Fim do procedimento da introdução.

1999 — Contém os números que definem os seis pontos diferentes.

2000/2010 — Marca o começo do procedimento das janelas. 2010 limpa o écran.

2020/2030 — Define uma janela pequena no princípio do écran, e escreve lá uma mensagem.

2040/2060 — Abre três janelas em diferentes partes do écran, todas com caracteres de tamanhos diferentes.

2070/2120 — Escreve 150 caracteres aleatórios, diferentes, nas 3 janelas, em cores diferentes (aleatórias).

2130/2230 — Escreve uma mensagem no canto superior esquerdo do écran, usando Mode 1 (para o tamanho dos caracteres).

2240/2290 — Faz andar a mensagem à roda no écran. Linha 2290 para a mensagem por uns segundos e depois limpa o écran.

2300 — Marca o fim do procedimento das janelas.

3000 — Marca o princípio do procedimento gráfico.

3010 — Deixa o écran branco e limpa-o.

3020 — Escreve uma mensagem no princípio do écran usando os caracteres em Mode 2.

3030 — Desenha 40 linhas aleatórias no écran.

3040/3050 — O écran fica todo vermelho. 3050 Pára o programa por alguns momentos.

3055/3120 — A cor fica a fazer Flash.

3130 — Mete o conteúdo da Rom no écran.

3140 — Põe o conteúdo da Rom no écran e vai passando gradualmente de verde para preto.

3150 — Limpa um certo espaço na RAM.

3155 — Define uma janela pequena na parte superior esquerda do écran.

3160 — Escreve uma mensagem, usando 64 caracteres por coluna.

3170 — As cores ficam a fazer Falsh.

3190 — Põe o conteúdo da janela em memória.

3200/3210 — Aumenta a mensagem, 3 vezes na direcção X e 6 vezes na direcção Y. A linha 3210 faz uma pausa no programa por alguns segundos e depois limpa o écran.

3220 — Marca o fim do procedimento gráfico.

4005/4007 — O écran fica vermelho, sendo depois limpo.

4010 — Escreve uma mensagem no princípio do écran em letras aumentadas.

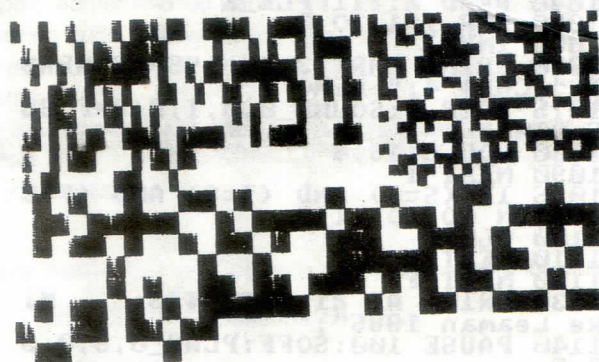
4020/4040 — Abre duas janelas sobrepostas.

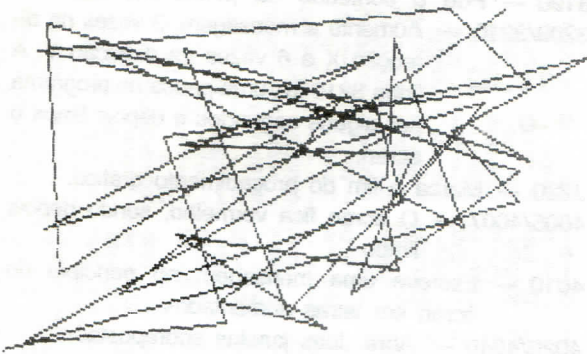
4050/4080 — Esta rotina faz rodar a mensagem na segunda janela.

4085/4090 — Roda a mensagem para a esquerda. 4090 pára o programa por alguns segundos.

4100/9999 — Assinala o fim do procedimento do écran Fim.

YS MEGABASIC DEMO
YS MEGABASIC DEMO
YS MEGABASIC DEMO
YS MEGABASIC DEMO
YS MEGABASIC DEMO
YS MEGABASIC DEMO
YS MEGABASIC DEMO

YS MegaBasic pode operar com 10 janelas em caracteres de dife-rentes tamanhos ,cada

As janelas podem-se dividir e rodar em todas as direccoes

Os graficos podem ser desenhados no ecran e exibidos pelo comando CHANGE.

As areas do ecran
poden ser aumentadas

As areas do ecran
poden ser aumentadas

FIM

```
  16 PCLEAR
  17 REPEAT
  20 INTRO
  30 WINDOWS
  40 GRAPHICS
  50 END
  60 UNTIL_0
1000 @INTRO
1003 MODE_4: CLS
1005 WINDOW_0,0,24,64
1010 BRIGHT 0: PAPER 0:MODE_4
1020 FOR F=0 TO 2
1025 FONT_F: RESTORE 1999
1030 FOR S=1 TO 6
1040 READ Z:STIPPLE_Z
1050 FOR I=1 TO 7
1060 INK I
1070 PRINT CHR$ 3;"   YS MEGABAS
IC";CHR$ 4;" DEMO "
1075 SREP_1:SOUND_0,0,I,2+2*F,25
5:SON
1080 VDU_2,13,4
1090 NEXT I
1095 IF (S=6) AND (I=8) AND (F=2
) THEN GO TO 1110
1100 CLW_3
1110 NEXT S
1120 NEXT F
1130 PRINT AT 21,0;CHR$ 3;"    Mi
ke Leaman 1985";
1140 PAUSE 100:SOFF:PLAY_0,5,0,3
0,5
1145 PRINT AT 21,0; PAPER 1; OVE
R 1;"
     ";
1150 CURRENT_4:WINDOW_21,0,2,28
1160 CURRENT_5:WINDOW_21,28,2,36
1170 FOR A=1 TO 80
1180 CURRENT_4:PAN_0,-2
1190 CURRENT_5:PAN_0,2
1200 NEXT A
1204 INPUT ;
1205 FOR A=1 TO 5
1210 FOR P=1 TO 7
1220 CHANGE_56,8*P
1230 PLAY_1,20,0,20,5*P
1240 NEXT P
1250 NEXT A
1260 ENDPROC_INTRO
1999 DATA 1,3,5,6,14,15
2000 @WINDOWS
2010 CLS
2020 CURRENT_2:WINDOW_0,0,5,64:
PAPER 1: INK 7:MODE_2:CLW_0
2030 PRINT "YS MegaBasic pode op
erar com 10 janelas ,  em caract
eres de dife-rentes tamanhos ,ca
da"
2040 CURRENT_4:WINDOW_5,0,8,40:M
ODE_3
2050 CURRENT_5:WINDOW_5,40,8,24:
MODE_2
2060 CURRENT_6:WINDOW_13,0,8,64:
MODE_4:STIPPLE_15
2070 FOR A=1 TO 150
2080 FOR W=4 TO 6
2085 CURRENT_W
2090 INK RND*7: PAPER 9
2100 VDU_(128+RND*15)
2105 IF A=150 THEN CLW_3
2110 NEXT W
2120 NEXT A
2130 FOR A=1 TO 100
2140 CURRENT_2:SCROLL_0,-1
2150 CURRENT_4:PAN_0,-2
2160 CURRENT_5:PAN_0,1
2170 CURRENT_6:SCROLL_0,1
2180 NEXT A
2190 WINDOW_0,0,22,64: BORDER 1:
 INPUT ;: PAPER 6: INK 1:CLW_0
2230 PRINT AT 1,2;CHR$ 1;"As jan
elas podem-se dividir";"      e r
odar em todas as direccoes"
2240 PLOT 0,171: DRAW 112,0: DRA
W 0,-24
2250 DRAW -112,0: DRAW 0,24
2260 PAN_0,120:SCROLL_0,-120
2270 FOR Z=1 TO 127:PAN_0,-1:SCR
OLL_0,1: NEXT Z
2290 PAUSE 50: CLS
2300 ENDPROC_WINDOWS
3000 @GRAPHICS
3010 BORDER 7: PAPER 7: INK 1:CL
W_0
3020 PRINT CHR$ 2;"Os graficos p
odem ser desenhadosno ecran e ex
ibidos pelo comando CHANGE."
3030 FOR A=1 TO 40: DRAW INK 7;(
RND*255)-PEEK 23677,(RND*130)-PE
EK 23678: NEXT A
3040 CHANGE_7,2
3050 PAUSE 50
3055 FOR Z=1 TO 8 STEP 8
3060 FOR N=1 TO 5
3070 FOR I=0 TO 7
3080 CHANGE_7*Z,I*Z
3090 PAUSE 10
3100 NEXT I
3110 NEXT N
3120 NEXT Z
3130 PUT_2,0,0,0,22,32:FADE_0
3140 PUT_2,0,0,0,22,32:FADE_6
3150 PAUSE 50: CLEAR 39999
3155 WINDOW_0,0,2,20
3160 PRINT CHR$ 1;"As areas do e
cran ""podem ser aumentadas";
3170 FOR A=1 TO 20: PAPER RND*7:
 INK 9:CLW_3: PAUSE 10: NEXT A
3190 GET_0,40000,0,0,2,10
3200 SPUT_40000,0,50,3,6,10,16
3210 PAUSE 50: CLS
3220 ENDPROC_GRAPHICS
4000 @END
4005 INK 2
4007 CLS
4010 SPRINT 0,0,4,5,"  FIM  "
4020 CURRENT_2:WINDOW_0,0,22,64
4040 CURRENT_4:WINDOW_5,0,17,64
4050 FOR Z=1 TO 50
4060 CURRENT_2:SCROLL_0,-1
4070 CURRENT_4:PAN_0,1
4080 NEXT Z
4085 PAN_0,-10
4090 PAUSE 100
4100 ENDPROC_END
9999 PAUSE 0
```

MEGABASIC SPRITE MAGIC

Adapt. por ISABEL CRISTINA

«Your Spectrum» — Julho, n.º 15

Mais uma demonstração das capacidades do utilitáio MEGABASIC. Ao longo do écran irão aparecer um número infinito de janelas, nas mais variadas posições.

Linha 1 — Esta linha marca o fim da Sprite. Deve usar este comando no princípio de qualquer programa que use Sprite.

10 — Esta linha limpa o espaço acima da RAMTOP
20/70 — Arranca com a imagem
75 — Desliga todas as Sprites
80 — Usa oito Sprites
90 — Iguala o endereço duma Sprite individual
95 — Verifica se a primeira Sprite está ligada
100 — Começa a escrever
110 — Põe o X e o Y nas posições de arranque
120 — Põe o X em movimentos aleatórios de Y
130 — A rotina de Sprites tem 4 imagens
140 — Muda de imagem depois de cada 4 interruptores
150 — Define o princípio da imagem dada para cada Sprite
160 — Anda para trás 8 vezes
165 — Liga todos os Sprite
170 — Altera a direcção aleatória
180 — Anda para trás e volta a repetir
200 — DATA para a imagem

```
  1 POKE 56894,255
 10 CLEAR 44855: LET P=44856
 20 FOR S=0 TO 3
 30 READ A,B,C,D
 40 POKE P,A: POKE P+1,B: LET P
=P+2
 50 FOR Z=1 TO 14: POKE P,C: PO
KE P+1,D: LET P=P+2: NEXT Z
 60 POKE P,A: POKE P+1,B: LET P
=P+2
 65 FOR Z=0 TO 3: POKE P+Z,6: N
EXT Z: LET P=P+4
 70 NEXT S
 75 POKE 56750,255
 80 FOR Y=0 TO 7
 90 LET S=56750+18*Y
 95 IF Y=0 THEN GO TO 110
100 POKE S,1
110 POKE S+1,115: POKE S+2,80
120 POKE S+3,5-RND*10: POKE S+4
,5-RND*10
130 POKE S+5,1: POKE S+7,4
140 POKE S+9,4: POKE S+15,6
145 POKE S+8,RND*3
150 POKE S+11,56: POKE S+12,175
160 NEXT Y
165 POKE 56750,1
170 LET S=56750+18*INT (RND*8)
180 POKE S+3,5-RND*10: POKE S+4
,5-RND*10
190 GO TO 170
200 DATA 255,255,129,129,63,252
,33,132,15,240,9,144,1,128,1,128
```

MEGAPLAY

Adapt. por Isabel Cristina
''Your Spectrum'' — n.15 Junho

Hoje vamos falar do comando PLAY. Tem sido bastante difícil trabalhar e entender este comando. A razão é clara, pode parecer um bocado ''ingénuo'' tentar converter os sons que agora o Spectrum é capaz de fazer, em palavras.
Que é que pensa disto?
Se introduzir o valor 12 no último parâmetro terá um som do tipo PHUST, se for o 9 terá PHIZZ.
Vamos apresentar um programa que demonstra as possibilidades do comando PLAY no YS MEGABASIC. Introduza este programa no MEGABASIC e depois faça RUN.
Aparecer-lhe-á um écran mostrando os parâmetros correntes do comando PLAY.
Use as teclas ↓ ↑ para escolher o valor que quer mudar, e depois aumente-o ou diminua-o com as setas ← →.
Quando tudo estiver pronto para ouvir o som que criou, carregue na tecla de espaço e aguarde com os ouvidos bem atentos.
Os parâmetros que escolheu também são disparados no écran.
Procedimentos:

Linha 1040-1100 — Chamam o procedimento principal.
2000-2170 — Desenha no écran.
3000-3200 — Desenha a informação nas janelas.
4000-4200 — Inicializa as variáveis principais do programa.
5000-5200 — Desenha no écran os parâmetros correntes.
6000-6200 — Vê o Keyboard e manda o programa executar outros procedimentos.
7000-8310 — Faz mover o cursor à volta e toca o som que foi criado.

```
PLAY_ EXPLORER           USE AS SETAS
                         PARA
                         SELECCIONAR
  A = 0     1            E MUDAR
  B = 0     255
                         ESPACO PARA
  C = 0     255            JOGAR
  D = 0     255
                         Q para SAIR
  E = 0     255
  ACTUAL    MAX          L.PRIVETT

PLAY       A    B    C    D    E
```

```
1000 REM by LA PRIVETTI
1020 REM MARCO      1985
1030 :
1040 PCLEAR
1050 SETSCREEN
1060 SETDISPLAY
1070 SETVALUE
1080 BOXIN
1090 SCANKEYS
1100 STOP
1110 :
1120 :
1130 :
2000 @SETSCREEN
2010 WINDOW 0,0,22,64
2020 PAPER 0
2030 INK 6
2040 BRIGHT 1
2050 OVER 3
2060 INVERSE 0
2070 MODE 2
2080 CLS
2090 DRAW 255,0
2100 DRAW 0,175
2110 DRAW -255,0
2120 DRAW 0,-175
2130 FONT 1
2140 LET A$="PLAY_ EXPLORER"
2150 SPRINT 20,10,1,2,A$
2160 INK 3
2170 ENDPROC_SETSCREEN
2180 :
2190 :
2200 :
3000 @SETDISPLAY
3010 LET A$="A B C D E"
3020 LET B$="= = = = ="
3030 INK 5
3040 DOWN 5,8,A$
3050 DOWN 5,12,B$
3060 FONT 2
3070 SPRINT 10,150,2,2,"PLAY"
3080 PRINT AT 2,39;"USE AS SETAS
"
3090 PRINT AT 3,39;"PARA"
3100 PRINT AT 4,39;"SELECCIONAR
"
3110 PRINT AT 5,39;"E MUDAR"
3120 INK 6
3130 PLOT 0,36
3140 DRAW 255,0
3150 INK 6
3160 PRINT AT 8,39;"ESPACO PARA"
3170 PRINT AT 9,39;"   JOGAR"
3180 BEEP .1,0
3200 ENDPROC_SETDISPLAY
3210 :
3220 :
3230 :
4000 @SETVALUE
4010 DIM M(5)
4020 DIM V(5)
4030 DIM P(5)
4040 FOR F=1 TO 5
4050 LET P(F)=3+F*2
4060 LET V(F)=0
4070 NEXT F
4080 RESTORE 4180
4090 FOR F=1 TO 5
4100 READ Q
4110 LET M(F)=Q
4120 PRINT AT 3+F*2,16;V(F)
4130 PRINT AT 3+F*2,24;M(F)
4140 NEXT F
4150 LET PRE=1
4160 LET POS=1
4180 DATA 1,255,255,255,255
4200 ENDPROC_SETVALUE
4210 :
4220 :
4230 :
5000 @BOXIN
5010 PLOT 20,80: DRAW 111,0
5020 DRAW 0,79: DRAW -111,0
5030 DRAW 0,-79
5040 PLOT 95,0: DRAW 0,36
5050 PLOT 95,19: DRAW 160,0
5060 MODE 1
5070 PRINT AT 18,26;"A          B"
;
5080 PRINT "     C          D";
5090 PRINT "     E"
5100 INK 4
5110 PRINT AT 15,25;"MAX"
5120 PRINT AT 15,14;"ACTUAL"
5130 MODE 2
5140 INK 2
5150 PRINT AT 12,40;"Q para SAIR
"
5160 INK 6
5170 PRINT AT 15,40;"L.PRIVETT"
5200 ENDPROC_BOXIN
5210 :
5220 :
5230 :
6000 @SCANKEYS
6010 LET Z$="                    "
6020 OVER 1
6030 PAPER 1
6040 INVERSE 1
6050 PRINT AT P(1),6;Z$
6060 REPEAT
6070 LET FINISH=0
6080 IF IN 63486=239 THEN LEFT
6090 IF IN 61438=251 THEN RIGHT6
100 IF IN 61438=247 THEN UP
6100 IF IN 61438=247 THEN UP
6110 IF IN 61438=239 THEN DON
6120 IF IN 64510=254 THEN QUIT
6130 IF IN 32766=254 THEN NOISE
6150 BEEP .005,-16
6160 UNTIL FINISH
6200 ENDPROC_SCANKEYS
6210 :
6220 :
6230 :
7000 @DON
7010 LET POS=POS+1
7020 IF POS>5 THEN LET POS=5
7030 POSITION
7040 ENDPROC_DON
7070 :
7080 :
7090 :
7100 @UP
7110 LET POS=POS-1
7120 IF POS<1 THEN LET POS=1
7130 POSITION
7140 ENDPROC_UP
7170 :
7180 :
7190 :
7200 @POSITION
7210 PRINT AT P(PRE),6;Z$
7220 PRINT AT P(POS),6;Z$
7230 LET PRE=POS
7240 ENDPROC_POSITION
7250 :
7260 :
7270 :
7300 @LEFT
7310 LET S=POS
7320 LET V(S)=V(S)-1
```

```
7330 IF V(S)<0 THEN LET V(S)=0
7340 OVER 0
7350 PRINT AT P(S),16;V(S);" "
7360 OVER 1
7370 ENDPROC_LEFT
7380 :
7390 :
7400 @RIGHT
7410 LET S=POS
7420 LET K=M(POS)
7430 LET V(S)=V(S)+1
7440 IF V(S)>K THEN LET V(S)=K
7450 OVER 0
7460 PRINT AT P(S),16;V(S);" "
7470 OVER 1
7480 ENDPROC_RIGHT
7490 :
7500 :
7510 :
8000 @QUIT
8010 LET FINISH=200
8020 PAPER 0
8030 INK 7
8040 OVER 0
8050 CLS
8060 ENDPROC_QUIT
8070 :
8080 :
8090 :
8100 @NOISE
8110 LET A=V(1): LET B=V(2)
8120 LET C=V(3): LET D=V(4)
8130 LET E=V(5):MODE_1
8140 OVER 0
8150 INK 7
8160 PAPER 0
8170 PRINT AT 20,26;A
8180 PRINT AT 20,32;B
8190 PRINT AT 20,38;C;" "
8200 PRINT AT 20,46;D;" "
8210 PRINT AT 20,54;E;" "
8220 MODE 2
8250 PAPER 1
8260 OVER 1
8270 INK 6
8280 PLAY A,B,C,D,E
8310 ENDPROC_NOISE
```

POKES PARA OS JOGOS

CHUCKIE EGGS

Primeiro faça: MERGE"": POKE 24501, 195: GOTO 1. Accione o PLAY no gravador. O jogo começará a ser lido normalmente, mas enquanto está a ser lido, observará uma mensagem de OK. Poderá agora juntar as rotinas abaixo indicadas. Para fazer RUN escreva RANDOMIZE USR 42000.

POKE 42837,0 — Dá-lhe vidas infinitas.

POKE 39151,0 — Nunca lhe aparece um bónus decrescente.

O programa a seguir, faz com que o CHUCKIE EGG trabalhe com Joystick da Kempston:

10 DATA 39277,103,39321,71,40311,71,40893,71,39341,79, 40254,79,40850,79,40655,95,40713,87.
40850,79,40655,95,40713,87.
20 RESTORE
30 FOR I = 1 to 9
40 READ A,B
50 Poke 1,219: Poke A-1, 253: Poke QA-2,0: Poke A-3,0: Poke A-4,203 Poke A-5B
60 NEXT I

O écran fica 672 bytes mais longo. Os valores dos endereços são:

0 — Espaço em branco 170
1 — Lado direito da escada 171
2 — Lado esquerdo das escadas 172
3 — Ovo 173
4 — Comida 174
5 — Chão e paredes 175
168 — Campainha mais alta da gaiola 176
169 — Campainha mais alta da gaiola 177
178 — A gaiola
179
180
181

Existem 3 secções na gaiola cada uma de 4 caracteres

O programa seguinte encontra todas as aberturas no topo do ecran:

10 FOR F = 46000 to 51375 step 672
20 FOR G = 0 to 31
30 Poke G — F,5
40 NEXT G
50 NEXT F

O programa seguinte põe comida no princípio da coluna:

10 FOR F = 46032 to 51375 step 672
20 FOR G = 0 to 31
30 If Peek (F-G) = 0 Then Poke (F-G), 4
40 NEXT G
50 NEXT F

A comida pode ser mudada para ovos. Para tornar o jogo ainda mais fácil, mude o 4 para 5 na linha 30.

Endereços do écran.

Écran	Endereços
1	46000-46671
2	46672-47343
3	47344-48015
4	48016-48687
5	48688-49359
6	49360-50031
7	50032-50703
8	50704-51375

Alguns dos caracteres estão no endereço (poke 23675,x:Poke 23676,y)

x = 248:y = 132 escadas, ovos, comida, chão, palavras
x = 248:y = 133 caracteres alongados
x = 248:y = 135 caracteres pequenos
x = 48:y = 138 gaiola
x = 160:y = 138 palavras

SIR LANCELOT

10 Input "n. de vidas":X
20 For A = 50000 to 50024
30 Read B = Poke A,B
40 Next A

50 DATA
49,125,91,221,33,128,91,17,128,36,62,103,55,205,86,5,243,
62,X,50,38,92,195,8,92
60 CLS
70 Print "Load Main Headerless Block"
80 RANDOMIZE USR 50000

Faça andar o gravador até ao bloco headerless, faça RUN do programa acima indicado, quando vir a mensagem, accione o Play no gravador, isto vai-lhe dar x vidas.

FRED
Primeiro escreva LOAD "" CODE 26384, faça ENTER, e accione Play no gravador. Depois do programa ter entrado faça SAVE "FRED" CODE 34500,22650.
Accione ENTER e grave esta versão. Reproduza-a de novo com LOAD "" CODE 245000, faça ENTER e PLAY. Introduza os Pokes abaixo indicados e para fazer RUN faça:
RANDOMISE USR 30285
Poke 31175,0 — Deixa a última unidade de poder durar para sempre
Poke 30418,X: Poke 31592,X: Poke 44696,X — X é o número das unidades de Poder (máximo 240).

STOP THE EXPRESS
Antes de introduzir os Pokes, faça o seguinte:
Escreva MERGE " ", faça ENTER, e accione o PLAY no seu gravador. Faça CLEAR 25999: LOAD " " CODE, accione ENTER e PLAY. Pare o gravador e escreva POKE 48111,201: RANDOMISE USR 48096, accione ENTER. Pode, a partir de agora, introduzir os POKES. Para fazer RUN dos códigos, faça RANDOMISE USR 32768.
POKE 34464,183 : POKE 34926,183 : POKE 35257,0
— Vidas infinitas
POKE 35780,0 : POKE 39549,0 — Tempo ilimitado
POKE 40673,0 : POKE 40674,0 — POKE 49261,N — Arranque na carruagem N (Tenha cuidado porque alguns valores destroem o jogo). Se isto não funcionar junte POKE 40075,4.

MONTY MOLE
POKE 38004,0 — Vidas infinitas
POKE 36301,201 — Nenhum esmagador
POKE 35874,255 — Deixa-o cair a uma grande altura

SON OF BLAGGER
Fazer MERGE da reprodução e apagar as linhas 70, 80, 81, 82, 90.
Inserir linha 45: POKE 27278,0. Este Poke vai-lhe dar um menu (todas as rotinas).

PI-IN'ERE
Para vidas infinitas : MERGE " " o programa e escreva POKE 38151,0 antes de fazer LET 1 ÷ USR 24576.

MANIC MINER
Para vidas infinitas : POKE 35136,0 depois de ler o Basic entre 30 e 40.

GISA KISS
1) Merge Basic Loader
2) Faça EDIT depois de ler as linhas POKE 25386,255 (255 vidas)
3) Mude USR para 25005
4) Faça RUN do programa e accione o gravador

CHILLER
Poke 34025,0 — Não há perda de energia
Poke 39791,0 : Poke 40062,0 : Poke 40333,0 — Para o movimento dos Sprites
Poke 41165,0 — homem principal desaparece

SABRE WULF
Poke 43575,255 para vidas infinitas (trabalha no Spectrum Séries 2 com Interface 1)

TIR NA NOG
Poke 34202,200 — Sem bichos

TORNADO LOW LEVEL
Faça Edit destas linhas no Header para vidas infinitas: 20 DATA 55,62,255,221,33,0,64,17,156,191,205,86,5,62,0,50, 190,136,50,15,132,201,50
FOR N = 65423 TO 65444
RANDOMISE USR 65423
Os jogos da Ultimate usam Poke 23756,1 o que permite fazer o Edit do Header

LUNAR JETMAN
Poke 36965,0 ou Poke 36966,224: Poke 36945,3 para vidas infinitas.

SELECT LEVEL
Poke 43092,x — onde x é o nível 99

PI-BALLED
Poke 46441,0 — Para vidas infinitas

SABRE WULF
Faça MERGE no Header. Faça Poke 23756,1 : CLEAR 65536. Faça EDIT da linha e apague PRINT USR 23424. Junte a linha 10 que inclui os seus Pokes. Junte a linha 20 PRINT USR 23424.
Poke 43575,255 — Vidas infinitas para um jogador
Poke 45520,255 — Vidas infinitas para dois jogadores
Poke 45509,x — Número de vidas
Poke 41725,255 — Não há limites nas vidas que foram ganhas
Poke 44929,0 — Desaparecem os maus. Estes só voltam a aparecer quando a tecla FIRE é primida
Poke 44786,0 — O Sabresman fica indestrutível
Poke 45001,0: Poke 41011,0: Poke 31602,0: Poke 48560,0 — Continua azul, super rápido e não há monstros
Poke 39702,30 — Fica livre dos montros que se materializam
Poke 44685,186: Poke 44676,255: Poke 44677,30 — Fica sempre em Cyan

TRANZ AM
Pare o jogo depois do bloco principal ter passado. Faça Poke 25446,0: CONTINUE

TRANS AM
Poke 25448,0 — Vidas infinitas

BACKPACKERS GUIDE TO UNIVERSE
Para tirar uma cópia do guia na impressora ZX ou Alphacom.

1 Clear 65535: Load "" Screen$
2 Paper 0: Ink 0
3 Print at 10,0;
4 Load "" Code
5 For x = 29182 to 29196: Read a: Poke x,a: Next x
6 Poke 29160,21
7 Data 62,127,219,254,230,1,194,215,113,205,172,14,195, 215,113.
8 Randomise Usr 29126

Agora faça Load do guia e accione SPACE para fazer a cópia.

ANDROID 2
Depois de se ter matado 4 vezes, pode ir para a mina ao mesmo tempo que o monstro. Aí deverá perder outras 2 vidas, mas depois terá tempo e vida infinitas.

GHOSTBUSTERS
Interesses
Name Ghostbusters
Conta 00166605 ($100,000)
Name S
Conta 20203002 ($24,000)
Name Tang Billy
Conta 15570011 ($112,100)

CRYPT
Nível de códigos
1 — Carnell
2 — Software
3 — Beats
4 — All
5 — the
6 — others

CAVELON
Accione todas as teclas nas colunas QWERTY e ASDF, quando aparecer Hi Chris what do you want? accione um número de 1-6 para começar no nível que deseja. Mova alternadamente o homem no labirinto e escreva JS WILLY.

TIR NA NOG
Se for tocado por um bicho accione SYMBOL SHIFT 6 para recomeçar o jogo.

WILD WEST HERO
POKE 23821,x — X é o número de vidas (32 no máximo).

ZIP ZAP
POKE 53751,0 : POKE 53752,0 : POKE 53753,0 : POKE 54141,0 : POKE 54142,0 : POKE 54143,0 : POKE 54144,0 — Energia infinita

MR WIMPY
POKE 33693,0 — vidas infinitas
POKE 33509,x — X é o número de vidas
POKE 43105,0 — Pimentos infinitos
POKE 33721,x — X é o número de pimentos
POKE 33501,0 — Salta o primeiro estrado.

MAZE DEATH RACE
POKE 26730,0 : POKE 26731,0 : POKE 26659,0 : POKE 26690,0 : POKE 26771,0 : POKE 26772,0 — Trabalha com interface 1

KOSMIC KANGA
POKE 35136,x — Onde x é a altura para saltar
POKE 23994,x — X é o número de vidas
POKE 36212,0 — Vidas infinitas

ESKIMO EDDIE
POKE 24686,24 : POKE 24687,76 — Vidas infinitas

KOKOTONI WILF
POKE 43742,0 — Vidas infinitas
POKE 42214,x — X é o número de vidas
POKE 42117,2 — Muda alguns Sprites para vermelho, que assim não matam Wilf

LAZY JONES
POKE 56693,0 — Vidas infinitas

SPECTACLE
Palavra chave : "Everyone's a nervous wreck"

ANDROID
POKE 52249,24 : POKE 52250,32 : POKE 53897,0 — Vidas infinitas

MUTANT MONTY
POKE 54933,0 — Dá 256 vidas

RIVER RESCUE
POKE 33420,0 — Vidas infinitas para o jogador n.º 1
POKE 33452,0 — Vidas infinitas para o jogador n.º 2

ZAXXON (STAR ZONE)
POKE 48825,x — X é o número de vidas

GILLIGANS GOLD
POKE 52882,0 : POKE 52881,0 : POKE 52882,0 : POKE 52883,0 — Vidas ilimitadas

FINDERS KEEPERS
POKE 34252,0 — Vidas infinitas
POKE 30394,x — X é o número de vidas

MUTANT MONTY
POKE 54800,0 : POKE 54867,0 : POKE 54893,0 : POKE 55321,0 Estes POKES dão vidas infinitas

PROJECT FUTURE
POKE 27662,0 — Desaparecem os Aliens

COOKIE
POKE 28695,62 : POKE 28696,5 : POKE 28697,0 : POKE 28698,0 — Vidas infinitas

FALL UY
Faça a seguinte listagem para vidas infinitas:

```
10 CLEAR 24100
11 LOAD "" code
12 RANDOMISE USR 65100
13 LOAD "" code
14 POKE 44204,0
15 RANDOMISE USR 41200
```

HORACE AND THE SPIDERS
POKE 25142,0 — Nenhuma colina no nível 1
POKE 28773,0 — Nenhuma aranha no nível 1
POKE 29626,0 — Puxa a corda rapidamente
POKE 29720,0 — Vai para a outra corda quando quiser
POKE 30070,0 — Todas as aranhas morrem instantaneamente no nível 3
POKE 25215,0 — Colinas e mais colinas para masoquistas
POKE 29910,0 — Aranhas lentas no nível 3
POKE 30116,0 — Aranhas infinitas no nível 3
POKE 29626,52 — Para as cordas no nível 3

FREEZE BEES
POKE 34610,0 — Vidas infinitas

PI — BALLAD
POKE 44416,x — X é o número de vidas

PSYTRON
POKE 28625,0 : POKE 28626,0 — Combustível interminável
POKE 41098,17 : POKE 41099,32 : POKE 411001,1 : POKE 41101,0 — Homens infinitos
POKE 26142,62 : POKE 26143,255 : POKE 26144,0 — Oxigénio interminável

FRANK'N'STEIN
POKE 28277,x — X é o número de vidas

AH DIDDUMS
POKE 24942,x — X é o número de vidas

PYRAMID
POKE 44685,0 — Energia ilimitada

ARCADIA
POKE 25776,0 — Vidas infinitas

UNDERWURLDE
Use o programa KNIGHT LORE mas faça as seguintes alterações:
Linha 50 mude 41012 para 41008 e 41021 para 41017
Linha 30 mude para PRINT USR 26610
Linha 20 e POKE 59376,0 para vidas infinitas ou POKE 45019,201 para remover os espíritos

SKY RANGER
Códigos: (1) enter; (2) magic; (3) pilot; (4) stomp; (5) pairs; (6) event; (7) recap; (8) alibi

KNIGHT LORE
Se evitar que um objecto caia no quarto do feiticeiro, o feitiço não aparecerá, permitindo-lhe passar pelo quarto mais facilmente.

```
10 CLEAR 24831 : RESTORE : GOSUB 60 : GO SUB 60 : POKE 32000,61 : POKE 62178,0 : POKE 62258,201 : POKE 62410,251 : POKE 62411,201 : RANDOMISE USR 62374
20 POKE 53567,0
30 PRINT USR 24832
40 DATA 23926, 23309,221,33,64,156,17,14,4,62,255,55,205,86,5,201
50 DATA 41012,41021,33,113,158,17,48,242,1,172,1,175
60 READ A,Z: FOR P = A TO Z: READ D: POKE P,D : NEXT P : RANDOMISE USR A: RETURN
```

Use esta listagem a seguir ao header inicial.
POKE 50084,201 — Pára a metamorfose
POKE 50206,0 — Dá dias infinitos
POKE 49759,n — 0 número de objectos a apanhar.

ATIC ATAC

POKE 36519,0 — Vidas infinitas
POKE 35353,0: POKE 35362,0 — Para os drenos de energia
POKE 36571,0: POKE 39092,0 — POKE 37229,175: POKE 37260,175 — As portas abrem-se com mais frequência
POKE 25373,x — O x é o número de vidas
POKE 26075,0 — Permite que o teu foguetão descole apenas com um bloco de combustível
POKE 25020,0 — Vidas infinitas

TUTANKHAMUN

POKE 27783,0 — Vidas infinitas
POKE 34953,57 : POKE 34970,58 — Para mudar o quarto do começo
POKE 27783,0 — Imortalidade.

NOVO AMSTRAD CPC 664

A COMINFOR anunciou recentemente a muito próxima comercialização no mercado nacional do novo AMSTRAD CPC 664.

Depois do sucesso completamente inédito do CPC 464 (200.000 unidades vendidas na Europa no último trimestre de 1984) o novo modelo, que retoma no essencial as características do seu antecessor.

Entre as diferenças mais notáveis destaca-se antes de tudo a incorporação da Unidade de Disc Drive e Interface, que substitui o leitor de cassetes integrado no CPC 464, bem como um ainda maior número de instruções de Basic, e uma significativa modificação do aspecto exterior.

De facto, e para além da incorporação da Unidade de discos, as teclas cinzentas substituem as teclas pretas do CPC 464, enquanto que as teclas verdes, vermelhas e azuis escuras daquele modelo, são agora de azul claro. O bloco numérico (que já representava uma inovação em relação aos outros computadores pessoais correntes) passa agora também a trazer incorporado dez teclas de funções, bem como as teclas de cursor são também agora ainda mais fáceis de manejar. Quando ao BASIC existem também diferenças dignas de realce e que interessarão certamente a todos aqueles que procurem aprofundar ainda mais as qualidades deste poderoso P. C. e das quais destacamos:

a) O bloco numérico duplicado permite agora a definição de dez teclas como teclas de função.

b) O BASIC, que era já uma das apostas sólidas da AMSTRAD, oi mais alargado de modo a substituir rotinas que anteriormente tinham de ser programadas em Basic 1.0 ou em linguagem máquina. O conjunto de novas instruções e em particular as instruções gráficas tornam ainda mais atraente os AMSTRAD. (A instrução FILL que atribui uma cor a uma superfície fechada pré-definida, ou a instrução FRAME que sincroniza a escrita gráfica e as linhas vídeo; A instrução CURSOR que determina o aparecimento ao desaparecimento do cursor, ou a instrução COPY CHR$ que envia um caracter para a posição daquele, ou ainda a instrução ON BREAK CONT, que impede a utilização da tecla ESCAPE a meio da execução de um programa), e contribuem seguramente sempre e cada vez mais para o êxito comercial da AMSTRAD, LTD e seus representantes.

c) Finalmente a Unidade de Discos Incorporada e sem dúvida a maior modificação em relação ao CPC 464 em que o binómio redução de preço/redução de espaço funcionou de forma e a contento do Utilizador, mantendo-se por outro lado a possibilidade de ligação da segunda Unidade de Discos, bem como do já tradicional gravador. De facto a Unidade possui as mesmas características da Unidade de Discos acoplável aos CPC 464 com sistema de exploração em CP/M e o LOGO elaborado pela DIGITAL RESEARCH, com discos de três polegadas possuindo 180 k de cada lado, bem como uma ampla e confortável biblioteca de Utilitários. Os bons compiladores de BASIC e de PASCAL, assim como o DBASE II, irão permitir a abertura deste campeão de vendas (e sem dúvida aquele em que a relação preço/qualidade e de momento o melhor), a mercados de actividade profissional séria, podendo praticamente rivalizar com sistemas de médio-porte.

Quanto à SOFTWARE PROFISSIONAL disponível é já grande a sua base de apoio e conta com uma Biblioteca de mais de 20 programas profissionais, vindos de Inglaterra, à qual se junta já o produzido em Portugal sob a égide da Cominfor, em que se inclui a já tradicional CONTABILIDADE, CONTAS-CORRENTES, FACTURAÇÃO (já com o IVA), STOCKS, PESSOAL, E TRATAMENTO DE LIVROS EM BIBLIOTECAS.

Características técnicas do AMSTRAD CPC 664

Nome	Amstrad C.P.C. 664
Fabricante	Amstrad, LTD
Importador-Distribuidor	Cominfor, LDA
Microprocessador	.Z80
Capacidade em Ram	64K
Capacidade em Rom	32K

Resolução Gráfica: 24x40 e 320x200 em 4; 24x80 e 640x200 em 2; 24x20 e 640x200 em 16 e cores/27.

Teclado em QWERTY c/74 teclas e bloco numérico com teclas de função.

Interfaces: Saída paralela tipo Centronics para impressora; Saída para monitor verde ou a cores; Saída para leitor de cassetes; Saída para Joystick; Entrada para segundo Disc-Drive.

Periféricos: Unidade de discos de 180 k com dupla densidade; Monitor gráfico monocramático mu a cores.

Extensões (Opcionais): Segunda unidade de discos; Impressora paralela; Joysticks e Modulador (para televisão normal).

TIMEX APRESENTA TC 2048

A TIMEX PORTUGAL promoveu no passado dia 26 de Junho o lançamento mundial do novo Timex Computer 2048. Em reunião com os orgãos de comunicação social o responsável máximo daquela empresa. António Gomes, fez o balanço da actividade da Timex desde a sua instalação em Portugal, pondo especial ênfase na viragem às novas tecnologias e na aposta efectuada na microinformática.

O TC 2048, agora lançado pela Timex, é um Spectrum compatível que, como tal, pode utilizar os milhares de programas e os muitos periféricos já desenvolvidos para aquele

microcomputador. Destinado a substituir o Spectrum no mercado nacional, o TC 2048 apresenta já eliminados os problemas existentes naquela máquina, e consiste basicamente num 2068, utilizando gravador de cassetes em vez de cartridges. Sobre este aspecto apresentamos em separado um quadro comparativo do ZX Spectrum, TC 2048 e TC 2068.

O TC 2048 contém um interface para JOYTICKS (KEMPSTONE) que possibilita a utilização de uma infinidade de SOFTWARE sem recurso a interfaces para JOYTICKS. Tem saída directa VÍDEO COMPOSTO, para monitores de vídeo monocromáticos ou cromáticos e ainda acesso aos sinais RGB através do porto expansor, permitindo assim, a ligação de monitores vídeo de alta resolução.

O teclado do TC 2048 funciona pela actuação de teclas rígidas sobre um circuito impresso proporcionando uma melhor introdução de programas e informação.

O HARDWARE permite o acesso ao modo de DISPLAY de 64 colunas com uma matriz de 8x6 para cada caracter, desde que sejam introduzidas nos programas as rotinas de tratamento de écran convenientes.

De salientar que o TC 2048 vai acompanhado de um MANUAL em PORTUGUÊS escrito por especialistas no ensino de INFORMÁTICA possibilitando o fácil acesso à LINGUAGEM dos COMPUTADORES.

Algumas das inovações tecnológicas apresentadas pelo TC 2048 são de importância decisiva para o sucesso deste microcomputador. Destas alterações salientamos as seguintes:

— A introdução de novos circuitos integrados CMOS (de baixo consumo) especialmente desenhados pela TIMEX que permitiu reduzir o número de circuitos electrónicos para um total de 15 no TC 2048 (contando com o interface JOYSTICK) contra os anteriores 26 no ZX/SPECTRUM.

— A substituição das antigas memórias dinâmicas com três tensões de alimentação pelas memórias dinâmicas actuais de alta capacidade e que apenas requerem uma tensão de alimentação, vem aumentar drasticamente a fiabilidade dos novos computadores.

— Nova filosofia de Keyboard com teclas rígidas actuando sobre um circuito impresso.

— A simples incorporação de um interruptor e de um LED vai reduzir o número de avarias provocadas pela adição inadvertida de periféricos ao computador com este ligado.

Para o 2048 está a Timex a desenvolver um amplo conjunto de software aplicacional. Existem desde já aplicações (em floopy disk) de contabilidade, gestão de stocks, tratamento de texto, base de dados, programas para profissões liberais, etc., para além de um sistema de gestão e controlo através de leitores de código de barras e cartões magnéticos. No final do corrente ano estarão prontos para comercialização o curso de electrónica, agenda, e gestão da conta bancária, entre outros, estando ainda a decorrer negociações com a Digital Research com vista à introdução do CP/M.

Também durante este mês vai ser posta à disposição do público a nova impressora matricial Timex 2080, de 80 colunas e 7 tipos de impressão gráfica, ao preço de 59 990$00.

O Timex TC 2048 é comercializado ao preço de 29 990$00, com um ano de garantia, beneficiando ainda de uma campanha de trocas do Spectrum na sua aquisição.

Comparação entre microcomputadores

	ZXSPECTRUM	TC 2048	TC 2068
Microprocessador	Z80A	Z80A	Z80A
RAM Física	48K	48K	48K
RAM Utilizável	41472 Bytes	41472 Bytes	38652 Bytes Modo Spectrum 41472 Bytes
ROM	16K	16K	24K
Écran	24x32	24x32-24x64	24x32-24x64
Resolução	256x192	256x192-512x192	256x192-512x192
Som	Beep	Beep	Beep e gerad programável
Joysticks	Não	1 Joystick	2 Joysticks
SolidStat Software	Não	Não	Sim
Gravador	Vulgar	Vulgar	Vulgar
Disc Drive	Pode Suportar	Pode Suportar	Pode Suportar
TV	UHF 36	UHF 36	UHF 36
Vídeo Composto	Não	Sim	Sim
RGB	Não	Sim	Sim
Interruptor ON-OFF	Não	Sim	Sim
Cores	8 Cores + Bright = 15	8 Cores + Bright = 15	8 Cores + Bright = 16
Tec. Circ. Impresso	Não	Sim	Sim
Circuitos integrados	26	15	15
Tecnologia	Bipolar	C-Mos (VLSI)	C-Mos (VLSI)
Manual/Português	Não	Sim	Sim
Garantia	6 Meses	1 Ano	1 Ano

NOTA: A redução da RAM utilizável no **TC 2068** deriva do facto de a ele se poderem ligar periféricos inteligentes, bem como Bancos de RAM num Bus Expansion e ainda da inserção de novos comandos, possibilidade de duplicação da resolução gráfica (passar de 24x32 a 24x64), programação de som e incorporação de Joysticks.

Visto estas facilidades estarem englobadas no sistema operativo do **TC 068**, do ponto da programação, esta redução transforma-se numa **VANTAGEM**, dado que os 38652 Bytes disponíveis vão permitir executar programas mais extensos e elaborados do que o **ZXSPECTRUM.**

QUEM AJUDA

«Há um assunto que eu gostaria de «aprender» e por isso venho através das linhas da nossa revista solicitar a cooperação de algum «entendido». trata-se dos capítulos 5 e 6 do manual de disketes.

FICHEIROS — CANAIS — SCP's

Alguém me poderia informar sobre livros e artigos sobre o assunto, ou mesmo publicar alguma coisa sobre este asunto na nossa revista?

Naturalmente que se tratar de distes para o Spectrum muito melhor, pois vai de encontro aquilo que eu tenho.

Quero catalogar os meus selos e a capacidade do Spectrum é pequena para isso. Com o uso dos tais ficheiros, julgo poder utilizar a capacidade das diskettes já que não saturaria a capacidade do Spectrum. Isso é o que entendi.

Grato pela atenção

Contactar com Manuel Alves Braga
Telef. 7825317
Vila Nova de Gaia

LISSAGE

Este pequeno programa não faz mais do que desenhar uma série de figuras simétricas, dependentes de 5 parâmetros que são gerados aleatoriamente.
Portanto as figuras são sempre diferentes e podem ir desde um simples ponto (improvável mas possível) a um desenho complexo e quase tridimensional.

Para entendidos:

As figuras de Lissage são figuras cujas equações paramétricas são duas sinusoides cujas frequências são múltiplas de uma frequência base comum, esfasadas de um determinado ângulo.
Ix = Ax sen(nwt) — w é a frequência de base e não afecta a forma da figura.

I
Iy = Ay senb(mwt-0)

— Ax e Ay são as amplitudes nos eixos dos xx e yy
— n e m são números inteiros
— 0 e um ângulo de esfasamento

As figuras deste programa são criadas gerando os parâmetros Ax,Ay,n,m, e 0,plotando 108pontos da figura de Lissajous correspondente e unindo os pontos consecutivos por segmentos de recta.

Nota: Quando os parâmetros n e m forem pequenos (5) a figura resultante é muito próxima da verdadeira curva de Lissajous. No programa esses parâmetros estão nas variáveis wx wy. experimente fazer LET wx = 2:LET wy = 3:LET ax = 70:LET ay = 70:LET e = 0:GOTO 30

```
   5 GO TO 100
  10 DEF FN x(a,w,t)=128+INT (a*
SIN (w*t))
  20 DEF FN y(a,w,t,e)=88+INT (a
*SIN (w*t-e*PI/180))
  30 CLS : LET t=0
  40 LET x=FN x(ax,wx,0): LET y=
FN y(ay,wy,0,e): PLOT x,y
  50 FOR n=1 TO 108: LET t=t+i
  60 LET x1=FN x(ax,wx,t): LET y
1=FN y(ay,wy,t,e)
  70 DRAW x1-x,y1-y: LET x=x1: L
ET y=y1: NEXT n
  80 RETURN
 100 RANDOMIZE : LET i=PI/54
 110 LET ax=INT (50+76*RND): LET
 ay=INT (40+47*RND)
 120 LET wx=INT (1+254*RND): LET
 wy=INT (1+254*RND)
 130 LET e=INT (RND*180)
 140 GO SUB 10: PAUSE 200: GO TO
 110
```


NOVOS PROGRAMAS

CHUCKIE EGG II

Harry tem de fazer um ovo gigante. Para consegui-lo, tem de apanhar numa fábrica, coco, leite e açúcar. Oito de cada um destes ingredientes tem de ser deitado no recipiente correcto. As oito partes também podem ser encontradas numa fábrica. Quando tiveres completado um ovo, tens de levá-lo, para este ser despachado.

Poderás então começar o próximo ovo, mas desta vez aparecerão mais monstros para atrapalhar.
Pode utilizar o Joystick, ou, se preferires, podes definir as teclas.

O jogo contém a opção SAVE que pode ser usada a qualquer momento.

Podes seleccionar uma tecla para executar este comando. Este jogo é uma aventura /arcádia, portanto lembra-te que encontrarás items que podem ser movidos de um lado para o outro e que, quando usados podem ajudar-te nas situações mais complicadas.

Diverte-te!

ROCKY HORROR SHOW

Entra na mansão do Dr. Frank'n'Furter se te atreveres. Janet ou Brad, foram empurrados para a máquina Medusa, a diabólica máquina de fazer cimento do Dr. Frank'n'Furter. A tua missão é encontrares as 15 peças desaparecidas da máquina Medusa. As peças estão espalhadas por alguns dos quartos de dois andares da mansão. Só podes apanhar uma peça de cada vez. Quando apanhares uma peça leva-a para o estrado onde se encontra a máquina e coloca-a na máquina.
Os quartos contêm raios laser, símbolos, cogumelos mágicos e seringas hipodérmicas, portanto tem cuidado. Também existem personagens sobrenaturais com quem te defrontarás. Existe Riff Raff, o mordomo maníaco, que te servirá experiências electrificantes; Magenta que tirará as tuas roupas. Os outros personagens são:
Columbia; Rocky Horror, a criação do Dr. Furter, que aparecerá nos lugares mais esquisitos; e o motociclista Eddie. Alguns personagens matar-te-ão imediatamente, enquanto outros farão comentários sobre a vida e a morte.
Os gráficos são bastante bons e o jogo é muito divertido.

ATENÇÃO AO NOVO ENDEREÇO

CLUBE Z-80

AVENIDA DA BOAVISTA

280 - 6º

4000 PORTO